(2)

ONR

Real Time Distributed System Studies/Scenarios
(Final Report)

June 24, 1991

John Molini
Sharon Maimon
Pat Watson

IBM Corporation 250/060
9500 Godwin Dr
Manassas, VA 22110

This paper overviews previous years research and specifies in detail three computer systems functionally similar to subsets of existing U.S. Navy ship and submarine systems. The computer system descriptions represent no specific system either deployed, under development, or proposed.

For each system, the computer system performance requirements and computer equipment selection criterion are given. The intent is to provide meaningful test problems for university based computer science research of real time distributed systems. The distributed processing scenarios assume that the research testbed systems contain at least three processors.

1.0 Introduction

IBM's efforts in support of the Office of Naval Research's Real-Time System Initiative have been accomplished in two phases.[1] In phase 1 of the project IBM integrated its Real-Time Communication Network (RTCN) prototype LAN with Carnegie Mellon University's (CMU's) Advanced Real-Time System (ARTS) operating system (OS) at the host OS kernel to network interface unit OS kernel level. IBM also performed experiments using RTCN to quantify the benefits of having network support for global semaphores rather than using a centrally located process. A factor of 5 difference in performance was measured. Finally, IBM developed and performed initial test on software designed to predict network scheduling performance and to measure actual results. The schedulability prediction software has been provided to CMU and incorporated into the distributed version of CMU's Scheduler 1-2-3, a system response time engineering tool.

In phase 2 of this research effort the ARTS/RTCN interface was extended to the application level through a cooperative effort between IBM and CMU. The design of the ARTS OS Version 1.0 allows selection of any one of several communication protocol stacks including RTCN and Ethernet. This facilitates performance evaluation of these protocol stacks for the hard real-time application environment.

The importance of the RTCN research prototype is that RTCN and its predecessor (currently in use on AN/BSY-1) are the only LANs that have been explicitly designed from inception to support rate monotonic scheduling of communications both at the message and packet level throughout the protocol stack, not just at the media access level. In short, RTCN is the only fiber optic LAN that is available for prototyping and measuring the quantitative benefits associated with using rate monotonic scheduling across the entirety of a real-time distributed system.

IBM has provided an RTCN LAN to the Software Engineering Institute (SEI) in support of SEI's Real-Time Scheduling in Ada Project and in the near future IBM will provide another set of hardware to the ARTS project at CMU in support of joint NOSC, ONR, and IBM funded hard real-time distributed system research. This is an integrated systems approach to the problem including scheduling algorithm development, OS kernel development, and schedulability analysis and response time performance monitoring capabilities.

For a number of years, researchers in the academic community have asked for requirements related to Department of Defense (DoD) real-time system applications (especially, those researchers supporting the Office of Naval Research's Real-Time System Initiative). These requirements have generally been unavailable to academic researchers because of the classified nature of the projects concerned. Without some meaningful characterization of these types of systems academic researchers cannot be sure they are addressing problems relevant to the DoD/Navy community and the DoD/Navy community cannot benefit as it might from their efforts in the critical area of distributed hard deadline real-time systems. This paper attempts to address this problem, it provides distilled functional requirements for three synthetic real-time systems. These systems are functionally similar to those that might be found on future U.S. Navy surface ships and submarines. They include an acoustic sensor system scenario driven by highly periodic processing, a combat information center scenario driven by man machine interface processing, and a radar system scenario driven by threat loading.

The scenario specifications are intended to be used for real-time distributed systems architecture research. Software systems designed to meet the specifications given by these scenarios will exhibit the complexity and timing constraints that can be expected to be found in a variety of future Navy platforms. The scenarios are intended to be used as workload benchmarks for comparing algorithms for managing shared system resources in a distributed real-time environment. The specific systems described however are contrived. Their functional descriptions have been taken from open literature. This data represents no specific system either deployed, under development, or proposed, however, they are based on many years of experience in working with such systems and they are tailored to match the laboratory resources that might be available to university researchers. Because different researchers will be working with equipments having greatly varying capacities, the processor and communication loadings are scalable via key, scenario dependent, parameters such as the number of beams, the number of operators, the number of targets, the rate at which operators page displays, the rate at which new threats appear, etc.

These distributed real-time scenarios assume that the research testbed systems contain target application processors interconnected by a Local Area Network (LAN). Since each scenario has been parameterized so the communication and processing resource requirements can be scaled up or down, this allows researchers to find the utilization levels where response time failures (for individual tasks and messages) occur using their proposed scheduling technique(s). This "breakdown" utilization is a figure of merit which is useful for comparing results on similar or divergent testbeds, for the same or different scheduling algorithms, and across a variety of research teams.

A-1

[1] This research has been performed for the Office of Naval Research (ONR) as part of its Distributed Real-time Systems research contract with IBM under Office of Naval Research Contract N00014-88-C-0745

The organization of the paper is as follows. Sections 2, 3, and 4 each describes one of the three scenarios. There are two subsections for each scenario. The first subsection describes the operating environment, the mission, human operator, and interfaces, and criterion for selecting hardware. The second subsection gives the requirements (processing, response time, and data flow) by function, along with the size and rate of each required dataflow. Section 5 describes an implementation of the first scenario wherein the functional requirements are mapped onto a specific system architecture and schedulability analysis results are discussed. Using the example provided in section 5 other researchers can arrive at their own architecture for this as well as the other two scenarios.

In addition to this paper, a spreadsheet is available for each scenario. The spreadsheet provides the means for assigning response times to the messages and tasks running on the testbeds in a manner that meets the timing requirements for each scenario. For the first scenario this information is contained in the tables presented in section 5.

2.0 Submarine Passive Sonar

This section describes a submarine passive sonar system. The data processing elements involved are typical of those found in detection and tracking systems on submarines, surface ships, or airplanes.

2.1 A Passive Sonar System

Sonar equipment is used for determining the presence, location, or nature of objects in the sea from underwater sound the objects emit. (10) Active sonar transmits an acoustic signal which, when reflected from a target, provides the sonar receiver with a signal. Based on this received signal, detections and position estimates are made. Passive sonar bases its detection and estimation on sounds which emanate from the target itself. (3)

In passive sonar, and the receiving subsystem of active sonar, the received acoustic waveform from each hydrophone consists of one or more signals and background noise. The hydrophone converts the acoustic waveform to an electronic signal. The signals are amplified, filtered, sampled, and digitized in a signal conditioner. The digitized hydrophone outputs from the signal conditioner are combined by a digital beamformer to form a set of "beams". A beam increases the sound from the beam direction and reduces the sound from other directions. Beam data are then processed to obtain detection and estimation statistics. Based on the values of these statistics, the system decides where targets are located. Detected targets are tracked by modifying the beamformer parameters and "steering" a beam toward the target.

A detected target is also analyzed. Analysis will include classification, distinguishing a signal returned from a target with regard to the type of target that produced the signal. A target is classified by the signal frequency spectrum and dynamics (e.g., a school of fish versus a submarine) of its target signal.

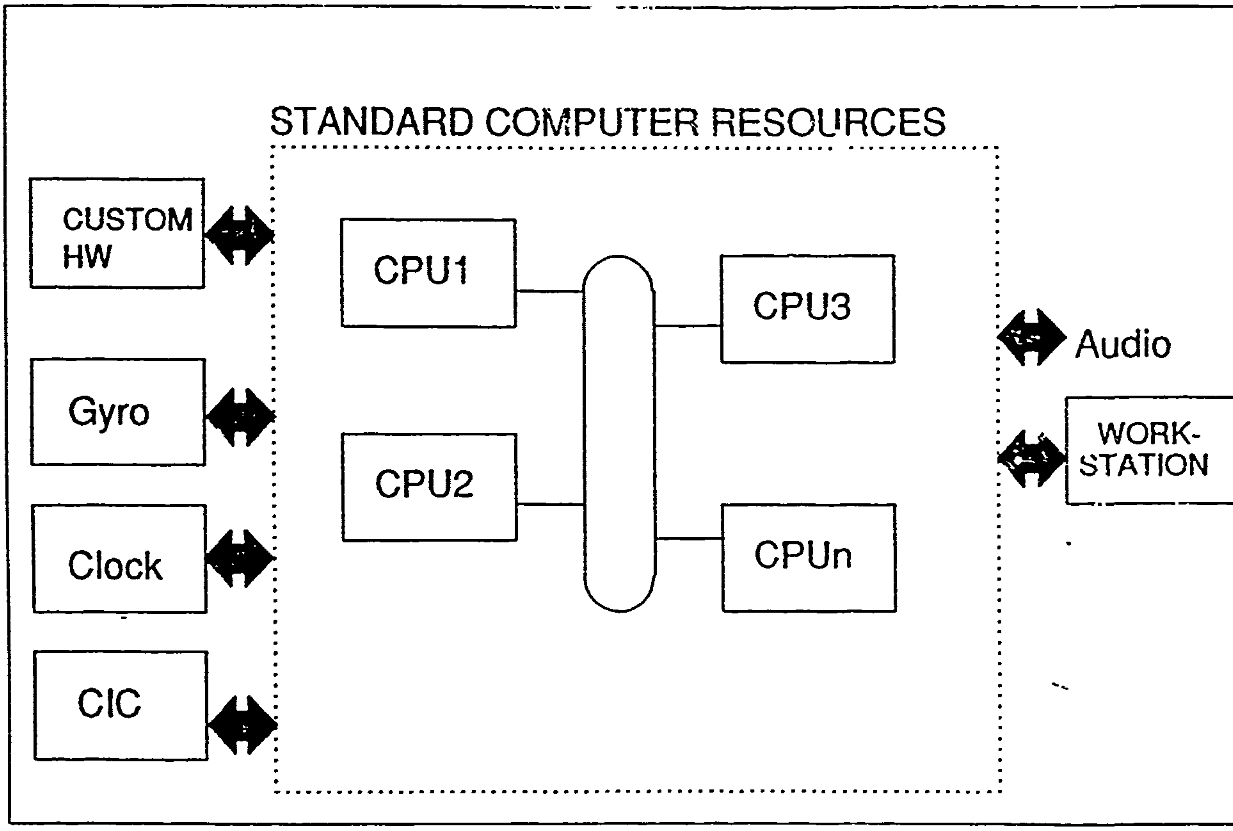

Figure 1. Submarine Sonar Processing Resources

The system described below is designed to detect and track submarine targets which can operate at any speed from 0 to 15 knots at any depth from 0 to 500 feet. The target sound source is assumed to consist of vibrational harmonics from rotating machinery with a nominal rotation rate of 2400 RPM. Therefore, it is expected that the returned signal from a submarine contains a fundamental 40Hz component and the first three harmonics.

The data processing load for a sonar system is roughly proportional to the number of hydrophones, number of beams, and the sample rate. The larger the array of hydrophones the greater the beamformer gain and the greater the detection range. The finer the beam width, the finer the display resolution which assists in resolving multiple targets even though they appear close together. Some of the beams are used for detection displays, some for tracking targets, and some just for listening. Typically the number of detection beams is fixed, so the detection processing load varies little over time. The number of steered tracker beams may change as targets come and go. When the number of tracker beams is changed, the tracking load changes proportionally.

Figure 1 shows the nominal configuration of standard computer resources. The Signal Conditioning and the Beamforming functions are performed in non standard, custom built hardware. This custom built equipment is channel attached to a standard processor. All of the standard processors within the system are networked together. External equipment is attached to standard processors by channels at points determined by the way the functions are partitioned. In this configuration, the analog audio output is assumed to be coming directly from the standard processors attached to the custom hardware. An alternative design with less cabling would be to use the digital data path for the audio data and perform the reconstruction at the workstations. This would require a high degree of clock synchronization between the workstations and the custom equipment in order to meet the timing requirements of section 2.2.8. In this configuration the timing requirement of section 2.2.8 must only be met within a single processor, not between processors connected by the LAN.

This configuration is chosen to fit the passive sonar system's mission and environment. Size and power are important criterion for selecting the processing equipment. Equipment spaces are usually cramped and it is difficult to run cabling. If there is a problem with the reactor, the system must operate on the limited power available from the ship's batteries. Power consumption also creates heat, heat requires cooling, and cooling can make acoustic noise. Equipment weight is not normally a significant constraint.

Another criterion is that the system should use as few unique equipment types as practical. This reduces the life cycle costs by reducing the number of different spare parts kept on hand and by increasing the production volume. The system should be able to run indefinitely, that is, it's availability (with a defined level of performance) should be near unity. Some performance degradation during maintenance time is permitted, but the entire system should never be powered off or reinitialized during normal operations.

In brief, the criterion for selecting preferred computer equipment are:

- The higher the processing density, the better.
- The smaller, the better.
- The less heat, the better.
- The fewer types, the better.
- The higher the availability, the better.

2.2 Data Processing for Passive Sonar

The time critical processing functions are shown in Figure 2. These include:

- Signal Conditioning
- Beamforming
- Detection
- Tracking
- Analysis/Classification
- Stabilization
- Time Synchronization
- Audio

The processing and I/O requirements are specified below. When this scenario is used for tests, the load on the computer resource may be varied by changing the performance parameters. These are:

- *N* - the total number of formed beams.
- *K* - the number of hydrophones
- *Nt* - the number of tracking beams.
- *Na* - the number of audio beams.
- *Nd* - the number of detection beams.
- *Nc* - the number of analysis beams.

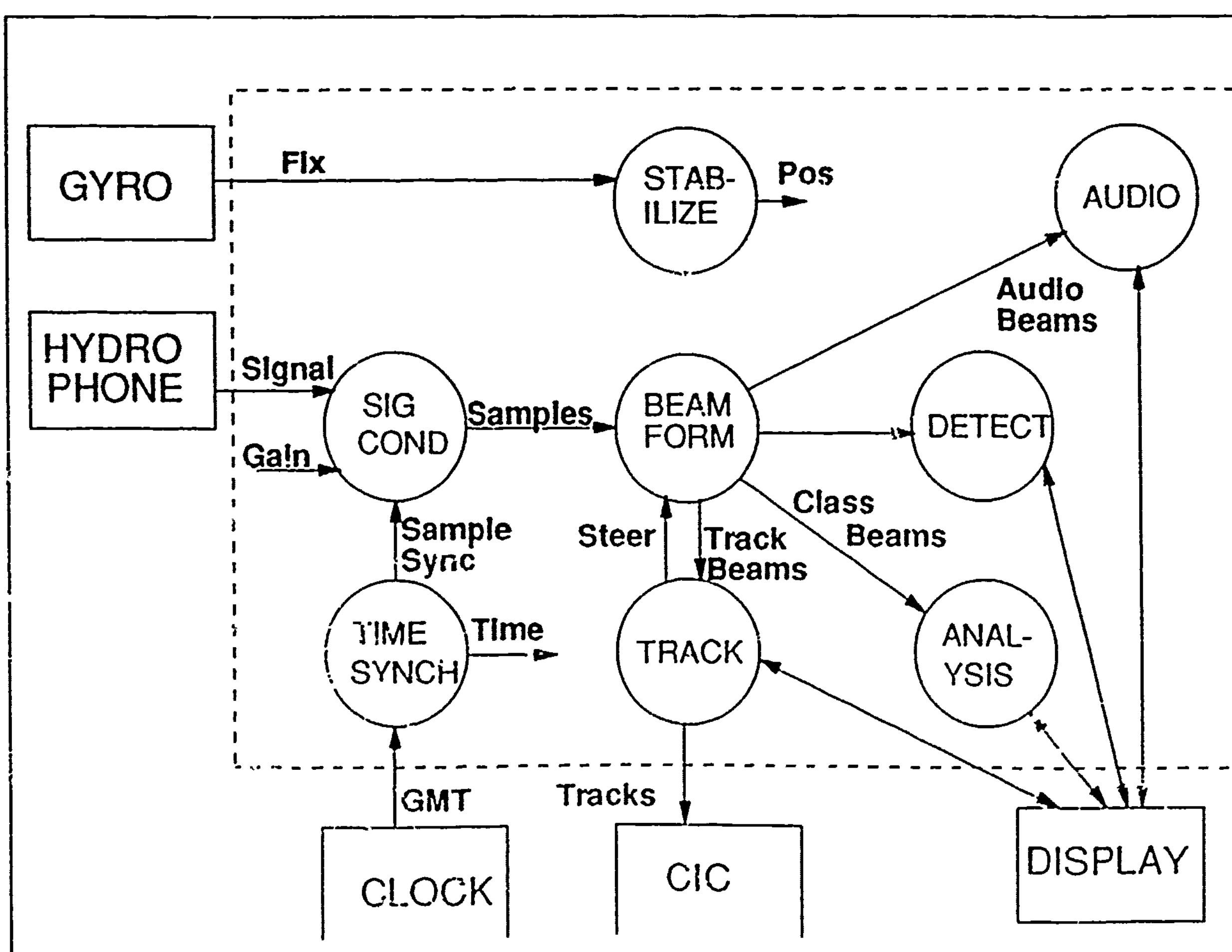

Figure 2. Passive Sonar Interfaces

in a real system, these parameters would be fixed and resources would be added until timing requirements were reliably met.

The response time requirements are given in terms of when data enters or leaves the system. A timing requirement stated for one function may imply timing constraints on other functions. Thus the timing requirement for the Track function limits the time the beamforming can take. Figure 3 shows the relationship between the functions and the system response time requirements.

2.2.1 Signal Conditioning: The Signal Conditioning function (SC) contains amplifiers, analog filters, automatic gain control, sample and hold circuits, and A/D converters.

SC amplifies, filters, and converts the hydrophones' signals into synchronous, discrete time samples. This produces the data and timing that drives the rest of the system. SC is performed in special purpose hardware that produces samples at 512 Hz. That is, the voltage amplitude is sampled from each hydrophone 512 times a second. (3)
See Figure 3.

2.2.2 Beamforming: The Delay and Sum Beamforming function (BF) delays the time samples from each hydrophone, scales and sums them. This increases the Signal to Noise Ratio (SNR) for signals from specific directions.

BF will stabilize beams by changing the beam delays to correct for sensor position and reflect the new direction of tracker beams. BF then sums the weighted samples to form the beams. All of the sensors are used to compute each beam. BF is also performed in special purpose hardware.

Input

- Time - Current time, date, and sampling pulse.
- Steer - Updated delay coefficients for aiming the tracker beams.
- Position - Location and attitude vectors of platform and sensors.

Timing Requirements BF computes beams fast enough to keep up with the Signal Conditioning. The beam samples are blocked into messages frequently enough to meet the timing requirements of Detection, Track, and Audio Reconstruction. The position data used to stabilize beams must be less than 0.5 seconds old. See Figure 3.

Output

- DetectBeams - Acoustic amplitude time samples for detection.
- TrackBeams - Acoustic amplitude time samples for tracking.
- AudioBeams - Acoustic amplitude time samples for listening.
- ClassBeams - Acoustic amplitude time samples for analysis.

2.2.3 Detection: The Detection function (DET) formats beams to allow an operator to find new signals.

Input

- DetectBeams - Acoustic amplitude time samples for detection.
- DetectSelects - Threshold selections and cursor position.
- Position - Location and attitude vectors of platform and sensors.

Processing DET compares the amplitude of the beams to selected thresholds, changing the data units to levels or quanta. This requantized data is integrated over time, filtered, and formatted for display. Only a portion of the information is displayed for every beam. Additional information about particular beams is provided to the display when the operator makes a selection with the display cursor. Processing will require 10*Nd thousand instruction per second (KIPS).

Timing Requirements Detection display data shall be sent at least four times a second. The display of detection data shall be synchronized to within 100 milliseconds of the Audio data. See Figure 3.

Output

- DetectDisplay - Time bearing plots of signal strength.

2.2.4 Track: The Track function (TR) steers beams to follow targets of interest.

Input

- TrackBeams - Acoustic amplitude time samples for tracking.
- Time - Current time, date, and sampling pulse.
- TrackSelects - Tracker assignments and display options.

Processing TR shall use the track beams to compute estimated bearing and bearing rate for assigned tracks. TR shall forward the results of these computation both as Track records and as plots. TR shall update the direction of the Track beams to follow the targets. The computations will require about 10*Nt KIPS.

Timing Requirements Track beam directions need to be updated to keep targets from being lost. Timely feedback is essential. Less than one second shall elapse from when a beam sample is taken, and the information from that sample is used to form subsequent beams. See Figure 3.

Output

- Tracks - Estimated bearing, bearing rate, and notations.
- TrackDisplay - Tracker location and quality formatted for display.
- Steer - Updated delay coefficients for aiming the tracker beams.

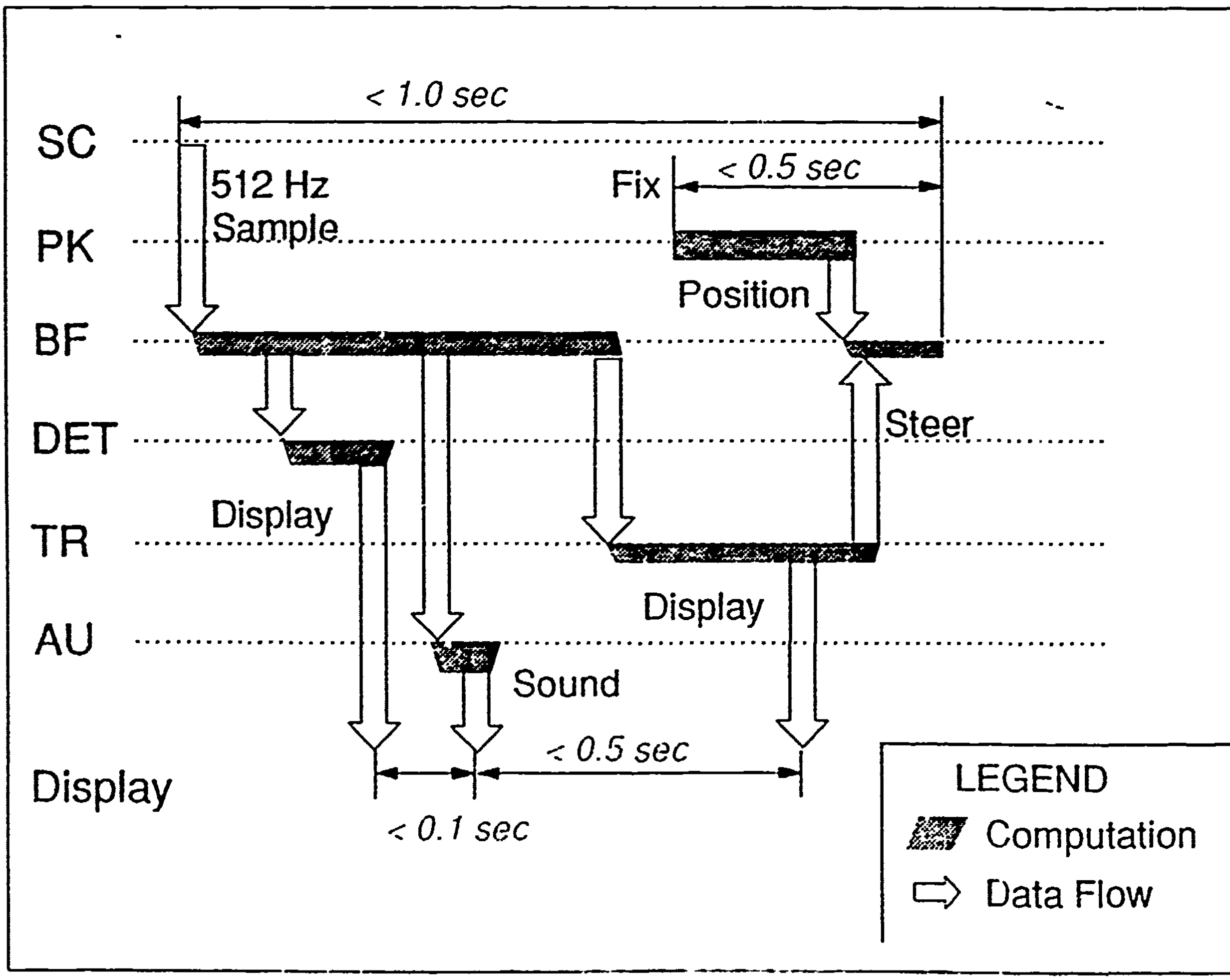

Figure 3. Front End Signal Processing

2.2.5 Analysis: The Analysis function (CI) identifies targets of interest.

Input

- ClassBeams - Acoustic amplitude time samples for analysis
- Time - Current time, date, and sampling pulse.
- Position - Location and attitude vectors of platform and sensors.
- ClassSelects - Display options and comparison choices.

Processing CL shall compare information from selected beams against known targets. Automatic comparisons shall occur continually. Automatic comparisons will take 50*Nc KIPS. Specific comparisons occur upon operator selection. Results from the comparison shall appear as annotations on Detection and Track displays. A single comparison will take 100 thousand instructions (KI).

Timing Requirements Results from specific comparisons shall be presented to the operator within 1 second of selection.

Output

- ClassDisplay - Annotations for the Detection and Track displays.

__2.2.6 Stabilization:__ The Stabilization function (PK) broadcasts the current location of the platform and hydrophones.

Input

- Fix - Navigational position, attitude, heading, and speed.

Processing PK shall compute attitude vectors for the platform and sensors. The resulting position data shall be sent to each of the other functions. Each computation will take 10 KI.

Timing Requirements Position information shall reach the functions within 200 milliseconds ˆ the fix being taken.

Output

- Position - Location and attitude vectors of platform and sensors.

__2.2.7 Time Synchronization:__ The Time Synchronization function (TS) provides the current time throughout the system.

Processing TS will superimpose the current GMT into the stream of timing pulses, and pass the resulting time information to each of the other functions.

Timing Requirements The delivered time signal shall be accurate enough to satisfy the Track re‹ uirements and steady enough to satisfy the Audio requirements. The timing pulse will reach the Signal Conditioning and Beamforming functions at 512 +/- 0.005 Hz.

Output

- Time - Current time, date, and sampling pulse.

__2.2.8 Reconstructed Audio:__ The Reconstructed Audio function (AU) provides high fidelity filtered audio signals along a beam of interest.

Input

- AudioBeams - Acoustic amplitude time samples for listening.

Processing AU shall reconstruct an analog signal from the beam data and present the signal to the operators headphones.

Timing Requirements. Figure 3 shows the timing requirements for the AU function. Synthesized audio shall be presented to the operator within 100 milliseconds of the corresponding display data. Since high fidelity reconstruction is essential, the reconstructed signal may slip no more than 0.10 seconds in five hours.

Output

- Sound - Analog signal for operator's headphones.

Table 1. The data flows			
Signal	**Description**	**Rate (Hz)**	**Size (Bytes)**
Samples	32 bit samples for each element each second.	512	3*K
Time	Pulse indicating when to take each sample. Maximum drift of +/- 10E-06.	512	1 bit
DetectBeams	Blocks of 32 bit beam samples for detection. 512 samples per beam per second.	TBD	TBD
TrackBeams	Blocks of 32 bit samples for Tracking. 512 samples per beam per second.	TBD	TBD
ClassBeams	Blocks of 32 bit beam samples for Analysis. 512 samples per beam per second.	TBD	TBD
AudioBeams	Blocks of beam samples for Audio. 512 samples per beam per second.	TBD	TBD
DetectDisplay	Waterfalling detection data. May be replicated.	4	2*Nd
Steer	Updated weights for the Track beams.	TBD	2*K*Nt
Tracks	Current status of the targets of interest.	> 1	64*Nt
TrackDisplay	64 bytes of light reading per Track beam per second. May be replicated.	TBD	TBD
ClassDisplay	4096 bytes of frequency data per analysis beam per second. May be replicated.	> 4	TBD
Position	Latitude, longitude, pointing and velocity vectors for the platform.	< 16	32
Time	GMT encoded onto the sampling clock.	TBD	8
Sound	High Fidelity analog waveform. May be replicated.	Analog	N/A
Signal	Analog waveform from each hydrophone.	Analog	N/A
Fix	Input from gyroscope.	16	24
GMT	Greenwich Mean Time from external clock.	1	24
Gain	Control settings for starting the signal conditioning.	Aperiodic	8000
DetectSelects	Input from the Detection operator. May be replicated.	> 10	8
TrackSelects	Input from Track operator. May be replicated.	> 4	24
ClassSelects	Input from Analysis operator. May be replicated.	< 1	32
AudioSelects	Input from the operator. May be replicated.	> 10	8

The data flows within the system are shown in Table 1. Redundant flows (for fault recovery), and system management data flows are not shown. In addition to this real time traffic, the system will have to support occasional file transfers of up to 10 Megabytes in length.

When multiple operators are using displays, each will have a separate set of headphones, input devices, and video surfaces. Data flows that have to be reproduced separately for each active operator are marked 'May be replicated'.

As long as the response time requirements are met, the blocking factors and message transfer rates for many data flows are fairly arbitrary. For these flows, the description shows how much data must be moved and the sizes and rates are marked 'TBD'.

3.0 Submarine CIC

This section describes a combat information center. The system is described as on a submarine but is typical of command and control or surveillance systems on submarines, surface ships, or land.

3.1 Combat Information Center

The information center provides a focus for all collected tactical information and platform status. The center typically has up to 20 workstations. Each workstation can access and display any or all of the system data. This data may include everything going on within hundreds of miles. All surface, air, and sub surface targets may be plotted against true geographical coordinates and topography. A full track history is kept for each target. All displays are kept current. (6)

The amount of data collected, its format (text, audio, or video), and the number of active operators determine the data processing load. Note that an "operator" may be a background processor applying classification or correlation algorithms to the arriving reports.

All the workstations in the CIC operate on the main track file. The track file contains the current information on the position and identity of all ships, aircraft, and submarines of interest to own ship. Information within the track file is updated many times a second. (5) A single function called Track Update fields the individual changes and distributes the current tracks to the workstations every two seconds.

Each workstation combines the current track information with data from the system database for display. Each display is either static or interactive. On a static display, the new reports are shown on a fixed background. On an interactive display, an operator is actively pulling more information from the system database as new tracks arrive. Many display formats are possible. Each format requires different data elements.

Many of the equipment selection criterion are a result of trying to make the operators as productive as possible. Display consoles must be responsive and comfortable. It should be easy to replace any of the system data with an update. It should be easy to allow a foreign system to "log in" and either read or deposit new information.

Space and power are important criterion. The system should use as few unique equipment part types as possible. This reduces the life cycle costs by reducing the number of different spare parts kept on hand. The system should be able to keep running for long periods of time. Some amount of scheduled maintenance time is permitted.

In brief, the criterion for selecting preferred computer equipment are:

- The more responsive each workstation, the better.
- The more that can be replaced, the better.
- The smaller, the better.
- The cooler, the better.
- The fewer part types, the better.
- The longer it operates, the better.

Figure 4 shows the configuration of standard computer resources for this scenario. The operators' display workstations are attached to the LAN. The other systems are channel attached to the standard processors at points determined by how the functions are partitioned.

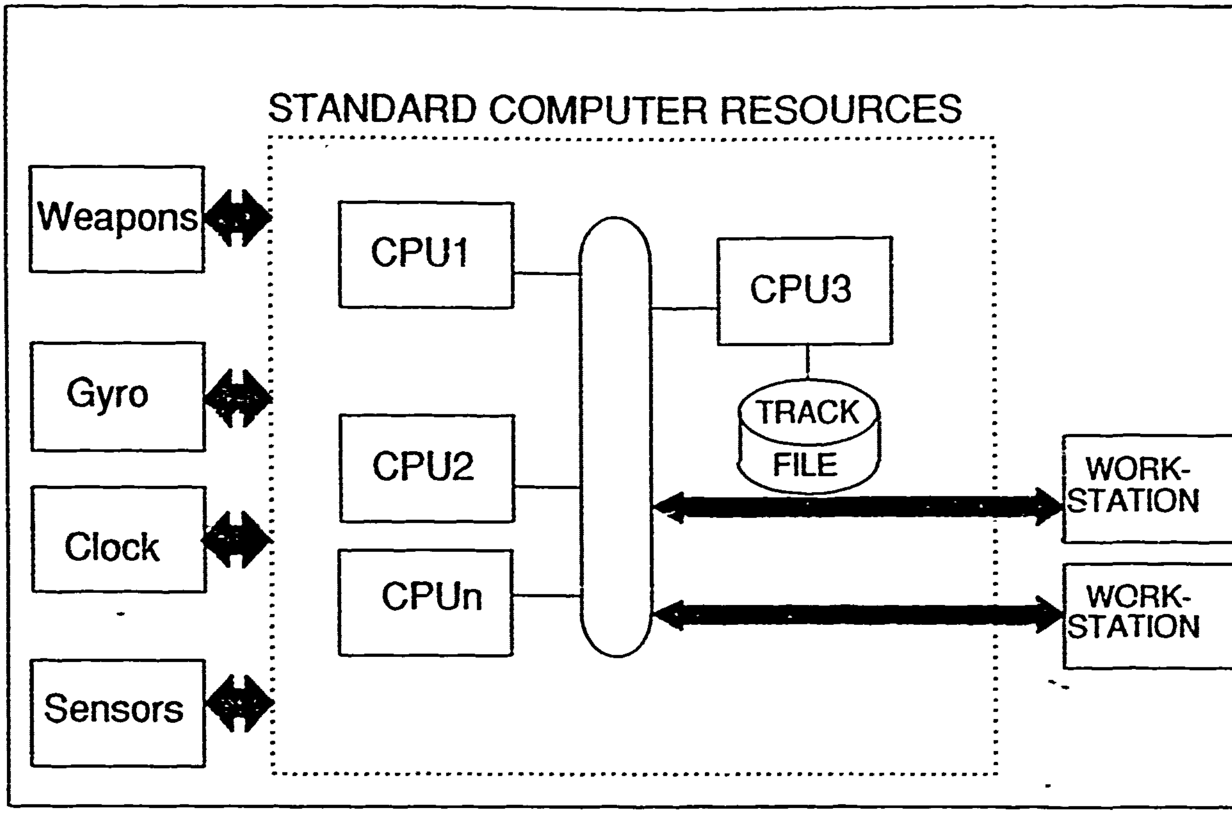

Figure 4. CIC Processing Resources

3.2 Data Processing for CIC

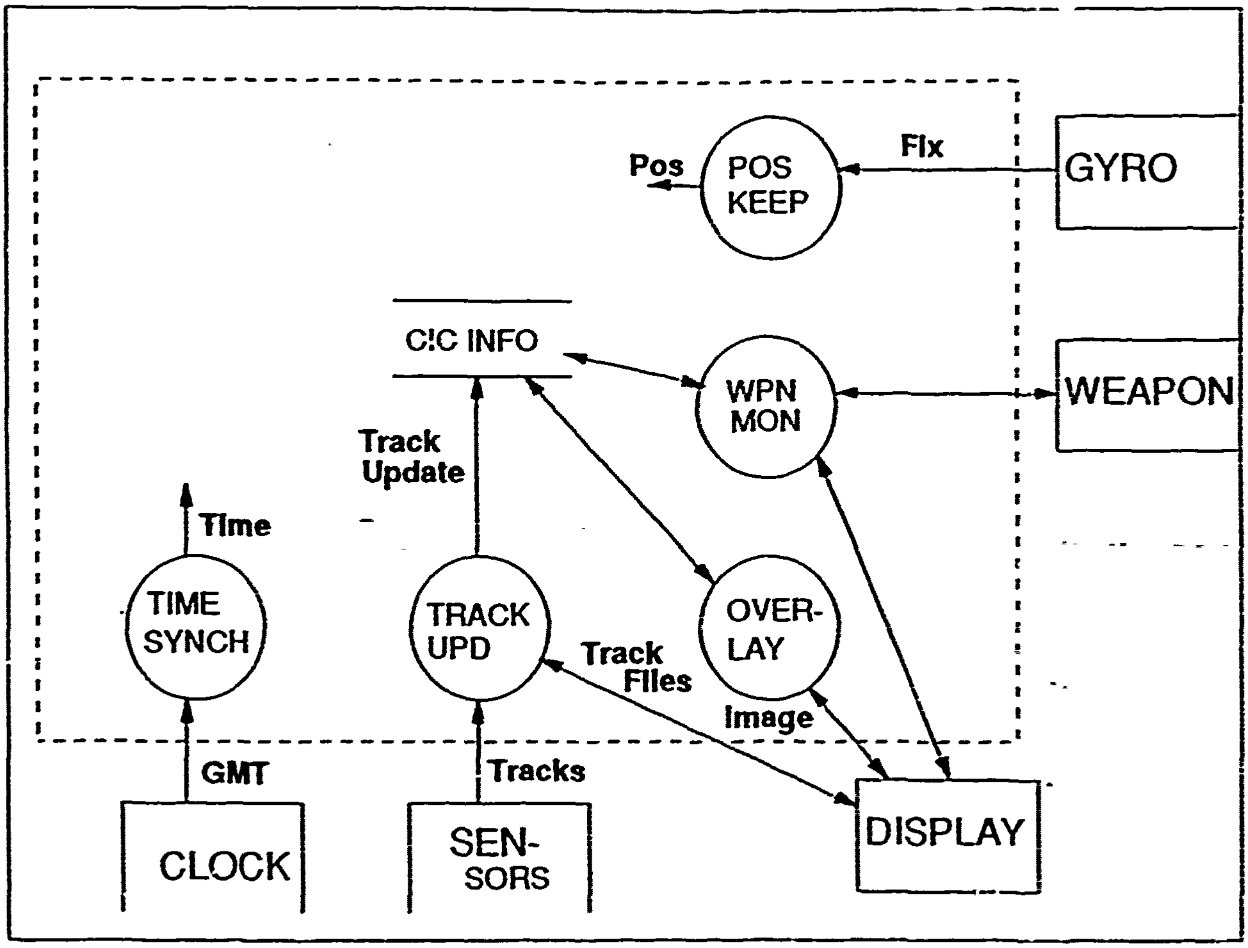

Figure 5. CIC Interfaces

The time critical processing functions include:

- Track Update
- Overlay Management
- Weapons Monitoring
- Position Keeping
- Time Synchronization

The databases are:

- CIC Information Repository

The processing and I/O requirements are specified in the following terms:

- *Ns* - the number of different sensor suites.
- *Nc* - the number of operators steering cursors. Moving the cursor around the screen generates a query for text information. The results appear in a pop-up window.
- *Np* - the number of operators paging. A page selection extracts an image from the database. The operator examines the image and (based on what is found) selects another page.
- *Rp* - the rate an operator flips through pages. The times an operator chooses to turn to a new page is a random variable.

<u>3.2.1 Track Update:</u> The Track Update function (TU) manages the collection and distribution of data on targets of interest.

Input

- Tracks - Track parameters from a single sensor
- Time - Current time and date
- Position - Location, velocity, and attitude vectors.

Processing TU shall collect tracks from the different sensor suites, reconcile different sensors tracking the same target, and apply these updates to the track file. TR shall distributed the updated track file on a regular basis. The processing will require 50*Ns KIPS. Updates to the track file must be serializable. While the track file might be distributed across several standard processors and updates applied to different parts of the track file concurrently, the interleaved updates must produce the same result as some serial application of those updates. (2)

Timing Requirements Tracks will be kept current enough to support the Overlay Management and Weapons Monitoring functions.

Output

- TrackFile - All current target tracks.
- TrackUpdates - Current position of targets of interest.

3.2.2 Overlay Management: The Overlay Management function (OV) extracts geographic and intelligence data from the system database for the displays.

Input

- PageSelect - Choice of new image.
- InfoSelect - Choice of new readout.
- Position - Location, velocity, and attitude vectors.
- Text Data - Information for a readout.
- Image Data - Information for a page display.

Processing OV shall retrieve and format system data. Any operator can view any system data (eg. any target in the track file). Preparing an read out will require 10 to 100 KI. Preparing a new page will require 20 to 40 KI.

Timing Requirements Readouts shall be returned to the display within 0.1 seconds. Page images shall be returned within 0.5 seconds. New tracks shall be less than 3 seconds old.

Output

- Readout - Pop-up text for the track picked by the display cursor.
- Image - Pixel data formatted for display.
- DataRequests - Navigational fix

3.2.3 Weapons Monitoring: The Weapons Monitoring function (WM) provides the data and timing needed by the weapons.

Input

- WpnSelect - Choice of weapon configuration.
- WpnStatus - Current state of each weapon.
- Targets - Current position of each target.

Processing WM shall pass configuration and aiming commands to the active weapons as directed by the operator's choices. WM shall provide current track data on operator selected targets. WM shall report the status of any active weapons to both the operators' workstations and the system database.

Timing Requirements Updated tracks shall reach the weapons within 0.5 seconds of being reported by the sensors. Status shall be displayed within 1.0 second of being reported by the weapons.

Output

- WpnDisplay - Display of weapon status and tracks.
- WpnOrders - Steering, Positioning, and configuration commands.
- WpnHistory - Current and projected state of each weapon.

3.2.4 Position Keeping: The Position Keeping function (PK) broadcasts the current location of the platform and hydrophones.

Input

- Fix - Navigational fix

Processing PK shall compute attitude vectors for the platform and weapons. The resulting position data shall be sent to the database.

Timing Requirements Position information shall reach the weapons within 200 milliseconds of the fix being taken.

Output

- Position - Location, velocity, and attitude vectors.

3.2.5 Time Synchronization: The Time Synchronization function (TS) provides the current time throughout the system.

Processing TS shall broadcast the current time and date to each of the other functions.

Timing Requirements Time kept by any two computers within the system shall differ by no more than 10 milliseconds. Time marks are received from the clock at a 1 Hz rate. If the standard processors are executing a time protocol such as NTP (7) the time need only be broadcast every few minutes since the system can allow several hours to pass for initial synchronization to become established.

Output

- Time - Current time and date.

Table 2. The Data Flows

Signal	Description	Rate (Hz)	Size (Bytes)
InfoSelect	Choice of new readout data.	2*Nc	50
DataRequest	Queries from Overlay Management for data.	2*Nc+ Rp*Np	100
Targets	Current target position and characteristics.	0.5	1K
WpnOrders	Weapons settings and steering commands	1	32
WpnDisplay	Current weapons status.	> 1	64
ImageData	Image retrieved from the database.	Rp*Np	1M
TextData	Textual data retrieved from the database.	2*Nc	100
WpnHistory	Running commentary on weapon status.	0.1	1K
WpnStatus	Current weapon state.	1	32
WpnSelect	Weapon display choices.	Aperiodic	50
PageSelect	Choice of new image.	Rp*Np	50
Fix	Input from gyroscope.	16	24
GMT	Greenwich Mean Time from external clock.	1	24
Image	Pixel data ready for display.	Rp*Np	1M
Position	Latitude, longitude, pointing and velocity vectors for the platform.	< 16	32
Readout	Text for cursor readout display.	2*Nc	50
Time	Greenwich Mean Time.	TBD	8
TrackFile	The entire current track file.	0.5	1M
TrackUpdates	Changes in the tracks from each sensor suite.	Ns	1K
Tracks	Updates to the tracks from each sensor suite.	TBD.	Ns*1K

The data flows within the system are shown in Table 2. Redundant flows (for fault recovery), and system management data flows are not shown. In addition to this real time traffic, the system will have to support occasional file transfers of up to 100 Megabytes in length.

As long as the response time requirements are met, the blocking factors and message transfer rates for many data flows are fairly arbitrary. For these flows, the description shows how much data must be moved and the size and rates are marked 'TBD'.

4.0 Surface Ship Radar

This section describes a phased array radar system. The system is described as shipboard. The system is typical of missile search, spacecraft tracking, and air defense systems, both ship and land based.

4.1 A Phased Array Radar System

A phased array radar uses a fixed antenna to project narrow beams of energy, each beam in a brief instant at a particular point. These pencil like beams (dwells) search particular volumes of space according to a computer controlled search plan. Signal processing is applied to the return from a single dwell to detect a target against background clutter and jamming. The information on detected targets are passed to data processing where tracks are developed and targets identified. (8)

When a track is classified as a threat and engaged, the radar search plan is modified to more frequently dwell on the target. This improves the quality of the track provided to the fire control system. The more targets detected and engaged, the more effective this system.(6)

In addition to air defense, this system can be used for air and sea traffic control and surveillance. For all of these tasks, the number of objects simultaneously tracked is important. The number of current tracks and the number of engaged targets largely determine the level of data processing load.

An anti-air combat system must fight under a wide range of conditions. A threat may be detected over the horizon by an off-board sensor. Threats may be high to low flying, fast or slow. The system may have to coordinate with other defensive mechanisms. Different tactics and rules of engagement may be in effect.

As described below, the combat system is defending a ship against low-flying mach 5 missiles using a counter missile with a minimum effective range of 4000 meters and a fly-out time of 4 seconds. For each threat, a shoot,look/shoot dectrine is assumed. The system has 15 to 27 seconds from detecting a missile to destroy it.

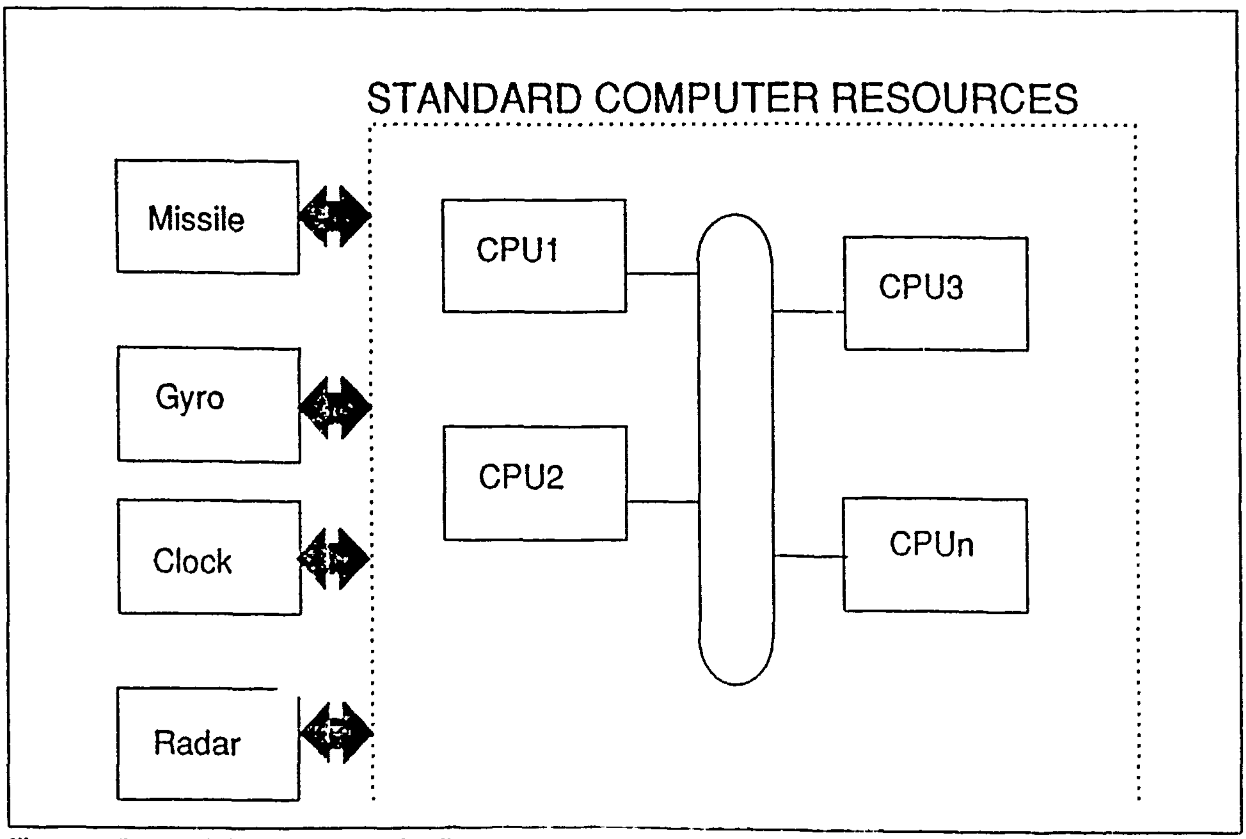

Figure 6. Surface Ship Radar Processing Resources

Figure 6 shows the configuration of standard computer resources for this scenario. In a real system there will be additional attachments for operator displays, but as described there are no time critical display processes and these connections are omitted. The radar equipment is assumed to be attached to standard processors through shared memory. This is done to meet the timing requirements of section 3.2.3.

Many of the equipment selection criterion are due to the system being installed on a surface ship. Since the radar array, missile magazine, and launcher are bulky, space and weight are not constraints on the computer equipment. The system should use as few unique equipment part types as possible. This reduces the life cycle costs by reducing the number of different spare parts kept on hand. The system needs to be able to survive combat damage. A shell or missile hit can destroy all the equipment within several meters. The system should be able to keep running for long periods of time. Some amount of scheduled maintenance time is permitted.

In brief, the criterion for selecting preferred computer equipment are:

- The more tracks, the better.
- The fewer part types, the better.
- The more that can be damaged without loss of critical functions, the better.
- The longer it operates, the better.

4.2 Data Processing for Surface Ship Radar

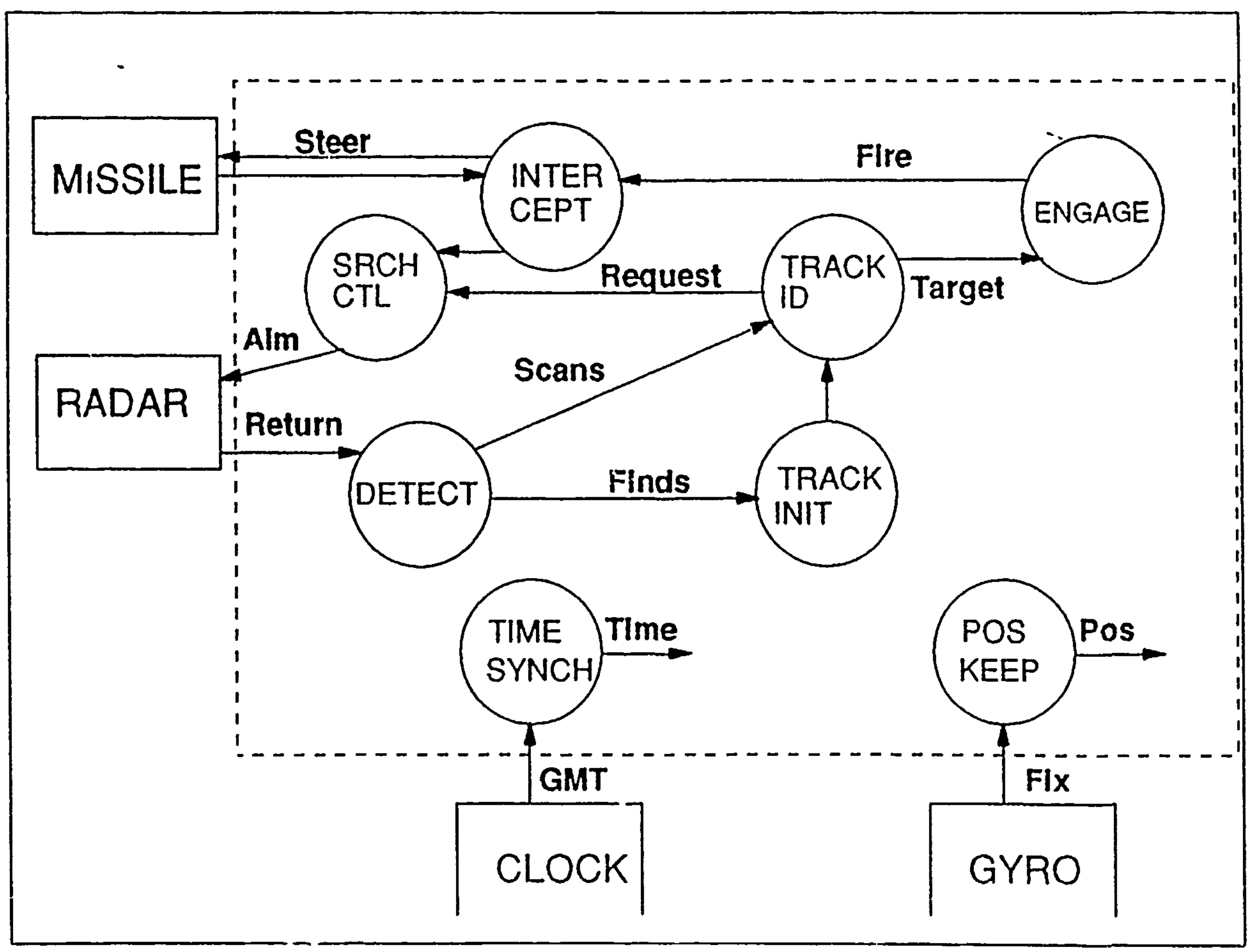

Figure 7. Surface Ship Radar Interfaces

The time critical processing functions include:

- Detect Track
- Track Initiation
- Search Control
- Track and Identification
- Engage Target
- Position Keeping
- Time Synchronization
- Intercept

The processing and I/O requirements are specified in the following terms:

- *T* - the number of active tracks.
- *H* - the rate hostiles are found.

These two terms are related random variables.

4.2.1 Detection: The Detection function (DET) thresholds the returns.

Input

- Returns - Amplitude of the completed radar dwells.
- Time - Current time and date.
- Position - Location, velocity, and attitude vectors.

Processing DET accepts and stores returns from each dwell. When enough successive returns have been accumulated from each range gate, filters are formed separating clutter signals from target signals. Processing requires 100 KIPS.

Timing Requirements. Thresholded dwells are blocked into messages frequently enough to satisfy the Track Initiation timing requirements.

Output

- Scans - Dwell return in the projected target direction.
- Finds - Dwell returns above a selected threshold.

4.2.2 Track Initiation: The Track Initiation function (TR) applies area thresholding.

Input

- Finds - Dwell returns above a selected threshold.
- Time - Current time and date.

Processing TR counts the number of reports above the detection threshold in a limited area. Large groups of point sources (e.g. bird flocks or decoys) are eliminated. This requires 100**H* KIPS

Timing Requirements TR eliminates spurious tracks from the set of targets within 30 seconds of initial detection. For some targets the analysis will produce ambiguous results. Up to 25 seconds worth of data may be needed to definitely eliminate a track. Thus, to meet this requirement the TR processing must be completed within 5 seconds. In addition, TR is producing tracks for Track Identification as shown in Figure 8.

Output

- Tracks - Targets to eliminate as spurious.

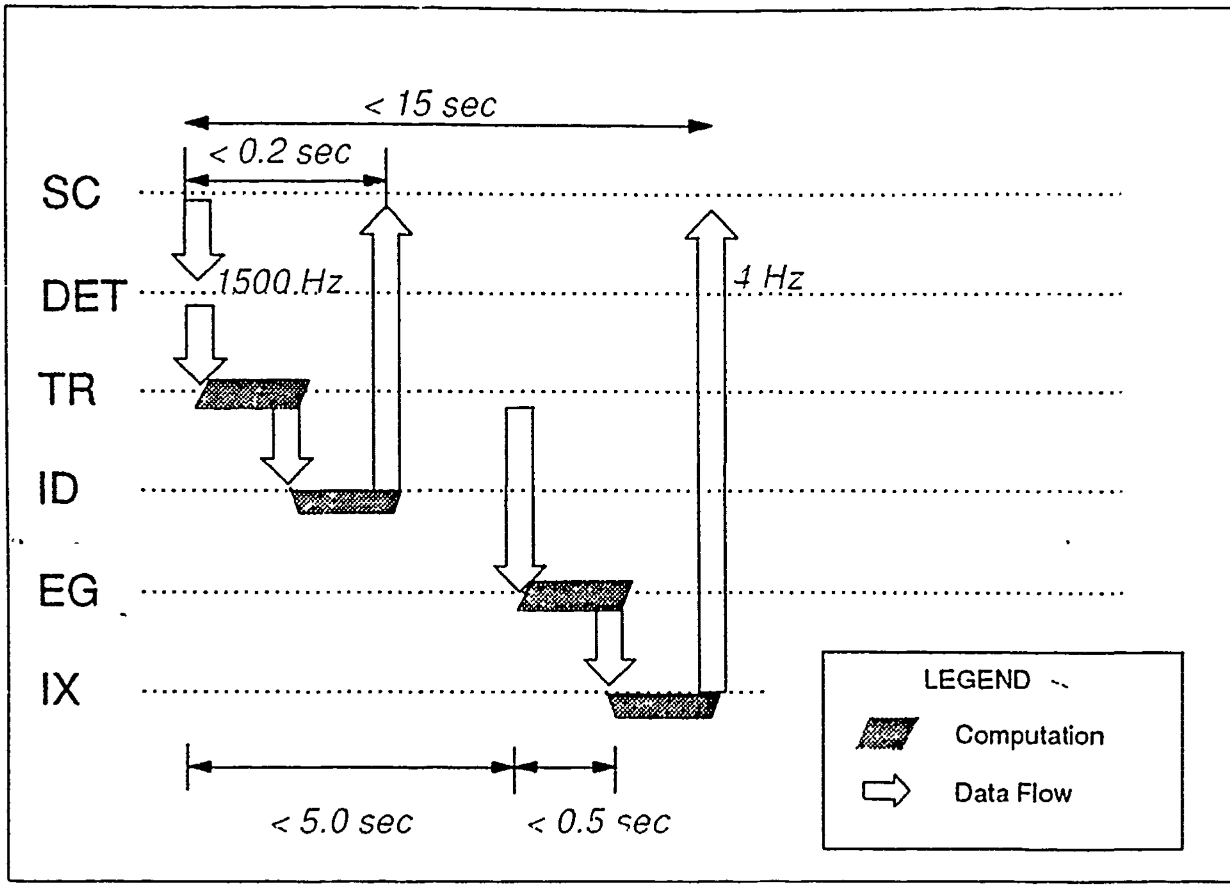

Figure 8. The Radar Scenario Timing Requirements

4.2.3 Search Control: The Search Control function (SC) schedules the radar dwells.

Input

- Requests - Request for a series of radar dwells.
- Position - Location, velocity, and attitude vectors.
- Time - Current time and date.

Processing SC accepts requests for radar dwells. The requests are built into a schedule of radar dwells. This schedule is passed to the radar. This will require 900 KIPS.

Timing Requirements Aiming instructions are calculated at 1500 Hz.

Output

- Aim - Schedule of upcoming radar dwells.

4.2.4 Track Identification: The Track Identification function (ID) estimates the motion of targets.

Input

- Scans - Dwell return in the projected target direction.
- Tracks - Targets to eliminate as spurious.
- Time - Current time and date.

Processing ID takes raw estimates of target position and computes smooth 1) present position, 2) present velocity, and 3) predicted position estimates. This requires 100* *T* KIPS.

Timing Requirements Request for dwells will be produced from data at most 0.2 seconds old.

Output

- Requests - Request for a series of radar dwells.
- Targets - Current track of hostile targets.

4.2.5 Engagement: The Engagement function (EG) decides which targets to attack.

Input

- Targets - Current track of hostile targets.
- Time - Current time and date.

Processing EN applies the current tactical doctrine. The threat posed by targets is evaluated. If automatic engagement is permitted, EN decides when to fire. This requires $8*T$ KIPS.

Timing Requirements EN will evaluate a new target's threat within 0.5 seconds.

Output

- FireOrders - Orders for launching missiles.

4.2.6 Position Keeping: The Position Keeping function (PK) broadcasts the current location of the platform and sensors.

Input

- Fix - Navigational fix

Processing PK shall compute attitude vectors for the platform and weapons. The resulting position data shall be sent to the database.

Timing Requirements Position information shall reach the weapons within 200 milliseconds of the fix being taken.

Output

- Position - Location, velocity, and attitude vectors.

4.2.7 Time Synchronization: The Time Synchronization function (TS) provides the current time throughout the system.

Processing TS shall broadcast the current time and date to each of the other functions. This will require 50 KIPS.

Timing Requirements Time kept by any two computers within the system shall differ by no more than 10 milliseconds.

Output

- Time - Current time and date.

4.2.8 Intercept: The Intercept function (IX) steers missiles to selected targets.

Input

- FireOrders - Orders for launching missiles.
- WpnStatus - Current state of each weapon.
- Time - Current time and date.
- Position - Location, velocity, and attitude vectors.
- Results - Radar returns from the missiles.

Processing IX configures and launches missiles as directed by the Engage function. In flight missiles are tracked. Terminal illumination is scheduled. Effectiveness is assessed. This requires $40*H$ KIPS.

Timing Requirements Requests for dwells to track in-flight missiles will be issued at 4 Hz.

Output

- Steer - Steering, positioning, and configuration commands.
- Requests - Request for a series of radar dwells.

Table 3. The Data Flows			
Signal	**Description**	**Rate (Hz)**	**Size (Bytes)**
Finds	Freshly detected flying objects.	10	100**H*
Scans	Dwell return in the projected target direction.	10	100**T*
Tracks	Intended sequence of locations.	1	200**H*
Aim	Schedule of upcoming dwells.	1500	400
Targets	Current hostile target tracks.	~2	100**T*
FireOrders	Orders to engage specific targets.	Aperiodic	64
Position	Latitude, longitude, pointing and velo[illegible] for the platform.	< 16	32
Time	Greenwich Mean Time.	~0.5	8
Requests	(from ID) Dwells to identify a target.	20	50**T*
Steer	Missile configuration and position commands	10	4000
Requests	(from IX) Terminal Illumination dwells.	4	400.
Results	Radar returns near launched missiles.	20	1000
Fix	Input from gyroscope.	16	24
GMT	Greenwich Mean Time from external clock.	1	~24
Returns	Digitized radar returns.	500	2000
WpnStatus	Current state of each weapon	20	4000

5.0 Submarine Passive Sonar Scenario Implementation

The submarine passive sonar scenario is implemented to execute under the ARTS operating system. ARTS provides "a predictable, analyzable, and reliable distributed real time computing environment." (9) The sonar application and ARTS operating system run on SUN 3/140[2] target machines. Network communications are provided by Ethernet or the Real-Time Communications Network (RTCN).[3]

The sonar application transmits messages between three CPUs. Messages that originate from or flow outside of the standard computer resources illustrated in Figure 1 are not transmitted. The allocation of processing to CPU is specified by the spreadsheet accompanying the sonar scenario description. Additionally, the spreadsheet defines the message rates, task rates and response times that fulfill the scenario requirements.

A computational entity under ARTS is called an artobject. (9) An artobject is implemented for each message sent or received within the standard computer resources. The artobject creates a single, periodic task (a thread) to send or receive the message and perform associated computation. Since the protocol being used to pass messages blocks execution of the thread receiving or sending the message, the one thread per message design was chosen for simplicity. Each receiving task is implemented by an ARTS server thread, each sending task by an ARTS client thread. The computation time of each thread is simulated by running a C Whetstone (1) benchmark program that has been modified to execute one thousand non-floating point instructions. Tasks that do not send or receive a message are implemented by combining their execution time with another process in the CPU running at the same rate. Message sizes and task computation time are parameterized to vary with system performance parameters.

Table 4, Table 5 and Table 6 describe the tasks (threads) running in each CPU, their timing requirements, messages and computation time. The task and message priorities are indicated as well. Priorities are assigned according to the rate monotonic (4) scheduling algorithm. A description of the table contents follows:

- "Task" contains a descriptive name of the task function.
- "KI/cycle" is the number of kilo instructions executed each periodic cycle by the task. If message size varies according to performance parameters, the formula is indicated. The values of the performance parameters used follow:
  - Number of track beams (Nt) = 5

---

2 Sun Microsystems, Inc.

3 IBM Corp.

- Number of audio beams (Na) = 6
- Number of detection beams (Nd) = 50
- Number of analysis beams (Nc) = 5
- Beamformer Rate (BF) = 4

• "Task resp" represents the amount of time the task has to complete its processing after it is started each cycle. The task response time is limited to the task period in this implementation, but in cases where the response time is actually larger than the period, the response time is listed in parenthesis.

• "Message" provides a description of the data.

• "Msg resp" provides the amount of time the message may take to be transmitted.

• "Task Rate" is the periodic cycle time of the task. This is also the message rate, if a message is being passed.

• "S/R (msg pri)" indicates whether the task is Sending or Receiving the message. The message priorities are listed in parenthesis for each send message for RTCN and Ethernet. Message priority is determined on a system basis based on the effective message rate. The effective message rate is the higher of the actual message rate or the message response rate. The fastest effective message rate is assigned the highest priority. Ties are arbitrarily broken. Message priorities for RTCN range from a high of 230 to a low of 30. Ethernet message priorities range from a high of 0 to a low of 7.

• "Size" is the message size in bytes. If the message size varies according to performance parameters, the formula is indicated. The performance parameters are listed above.

• "Priority" is the task priority. This value is computed based on the task rate and task response time using rate monotonic priority assignments. The task set is executed with the ARTS scheduling policy set to fixed. The task priorities for each CPU are computed independently. The highest priority is assigned to the task with the highest effective task rate. The effective task rate is the higher of the task rate or the task response time. Ties are arbitrarily broken. Thread priorities range from a high of 0 to a low of 127.

Table 4. Tasks in CPU 1

Task	KI / Cycle	Task resp (ms)	Message	Msg resp (ms)	Task Rate (hz)	S/R (msg pri)	Size (b)	Priority
Update Steering ***	24 (4*Na)	100 (200)	AudioSelects	100	10	R	43 (8*Na)	5
Specific Comparison	100	200	ClassSelects	300	Aper (.6)	R	32	10
Specific Comp 2	N/A	250 (300)	ClassDisplay	200	4	S (182/4)	5120 (1024 * Nc)	13
Estimate Tracks*	6.3 (5*Nt * BF)	250	Tracks	N/A	4	S (176/6)	80 (16*Nt)	12
Recompute Delays*	6.3 (5*Nt BF)	250 (500)	none		4			--
Form Sample 1 #	N/A	100	ClassBeams	250	4	S (180/5)	10 (2*Nc)	7
Form Sample 2 #	N/A	100	TrackBeams	250	4	S (178/5)	10 (2*Nt)	8
Format Display**	125 (10*Nd / BF)	250	DetectDisplay	200	4	S (184/4)	100 (2*Nd)	11
Rebuild Sound**	3 (10/BF)	250 (300)	none		4			--
Adjust Clock ##	3	125 (400)	Time	100	8	R	8	9
Receive New Fix ##	2	50	Position	100	8	R	32	2
Form Sample 3 ***	N/A	100	TrackSelects & DetectSelects	100	10	R	64 .	3

- * Estimate Tracks and Recompute Delays were combined into one task: Rate = 4 hz, KI = 10*Nt/BF.
- ** Format Display and Rebuild Sound were combined into one task: Rate = 4 hz, KI = (10*Nd/BF) + (10/BF).
- *** Update Steering and Form Sample 3 were combined into one task. Rate = 10 hz, KI = 24, Message Size = (8*Na) + 64, Priority = 3.
- # Form Sample 1 and Form Sample 2 were combined into one task: Rate = 4 hz, Message Size = (2*Nc) + (2*Nt), Priority = 7.
- ## Adjust Clock and Receive New Fix were combined into one task: Rate = 8 hz, KI = 5, Message Size = 40, Response = 50 ms, Priority = 2.

Table 5. Tasks in CPU 2

Task	KI / Cycle	Task resp (ms)	Message	Msg resp (ms)	Task Rate (hz)	S/R (msg pri)	Size (b)	Priority
Get Cursor**	12 (2*Na)	100	AudioSelects	100	10	S (206/1)	48 (8*Na)	4
Update Cursor**	10	100 (200)	TrackSelects	100	10	S (230/0)	48	3
Update Cursor**	16	100 (200)	DetectSelects	100	10	S	16	--
Comparison Request	5	100	ClassSelects	300	Aper (6)	S (174:?)	32	8
Display Comparison	5	100	ClassDisplay	200	4	R	5120 (1024 * Nc)	7
Automatic Comparison ***	62.5 (50*Nc / BF)	250	ClassBeams	250	4	R	10 (Nc*2)	10
Estimate Tracks ***	6.3 (5*Nt / BF)	250	TrackBeams	250	4	R	10 (Nt*2)	11
Show Det Display*	8	100	DetectDisplay	200	4	R	100 (Nd*2)	6
Show Track Display*	5	100	none		4			--
Adjust Clock	3	125 (400)	Time	100	8	R	8	9
Receive New Fix	2	50	Position	100	8	R	32	2

- * Show Det Display and Show Track Display were combined into one task: Rate = 4 hz, KI = 13.
- ** Update Cursor for track and detection and Get Cursor were combined into one task. Rate = 10 hz, KI = (2*Na)+36, Message size = (8*Na)+64, Priority = 3.
- *** Automatic Comparison and Estimate Tracks were combined into one task: Rate = 4 hz, KI = (50*Nc/BF)+(5*Nt/BF), Message size = (Nc*2)+(Nt*2), Priority = 10.

Table 6. Tasks in CPU 3

Task	KI / Cycle	Task resp (ms)	Message	Msg resp (ms)	Task Rate (hz)	S/R (msg pri)	Size (b)	Priority
Forward Tracks	5	100	Tracks	N/A	4	R	80 (Nt*16)	10
Build Time Message*	3	125 (400)	Time	100	8	S (186/3)	8	12
Adjust Clock*	3	125 (400)	none		8			..
Compute Attitude**	10	50	Position	100	8	S (188/2)	32	6
Receive New Fix**	2	50	none		8			..

- * Build Time Message and Adjust Clock were combined into one task: Rate = 8 hz, KI = 6.
- ** Compute Attitude and Receive New Fix were combined into one task: Rate = 8 hz, KI = 12.

Scheduler 1-2-3, a tool that can determine the schedulability of a task set (9) was run on each CPU load. The results are shown in Figure 9. The aperiodic tasks in CPU 1 and 2 were not included in the analysis, since a sporadic server is not available in our pre-release version of the tool. With the performance parameters specified, CPU 1 defines a 77% CPU load, CPU 2 defines an 82% load and CPU 3, 31%; all the CPUs are schedulable according to the rate monotonic closed form analysis capability of Scheduler 1-2-3. The task period used was the shorter of the task rate or the task response time. Execution times were based on the amount of time the KI routine required to run on the SUN 3/140. Times smaller than the clock granularity were assumed to be 1 millisecond per KI as a worse case, though tests showed the CPU time to be slightly faster.

Simulations run using Scheduler 1-2-3 showed approximately the same results as the closed form analysis when using the rate monotonic scheduling policy. Simulations showed that CPU 1 and CPU 2 were not schedulable if the FIFO scheduling policy was used. Figure 9 shows the CPU utilization for each CPU as computed by the closed form analysis and the simulations. The number of missed deadlines is indicated for the FIFO simulations.

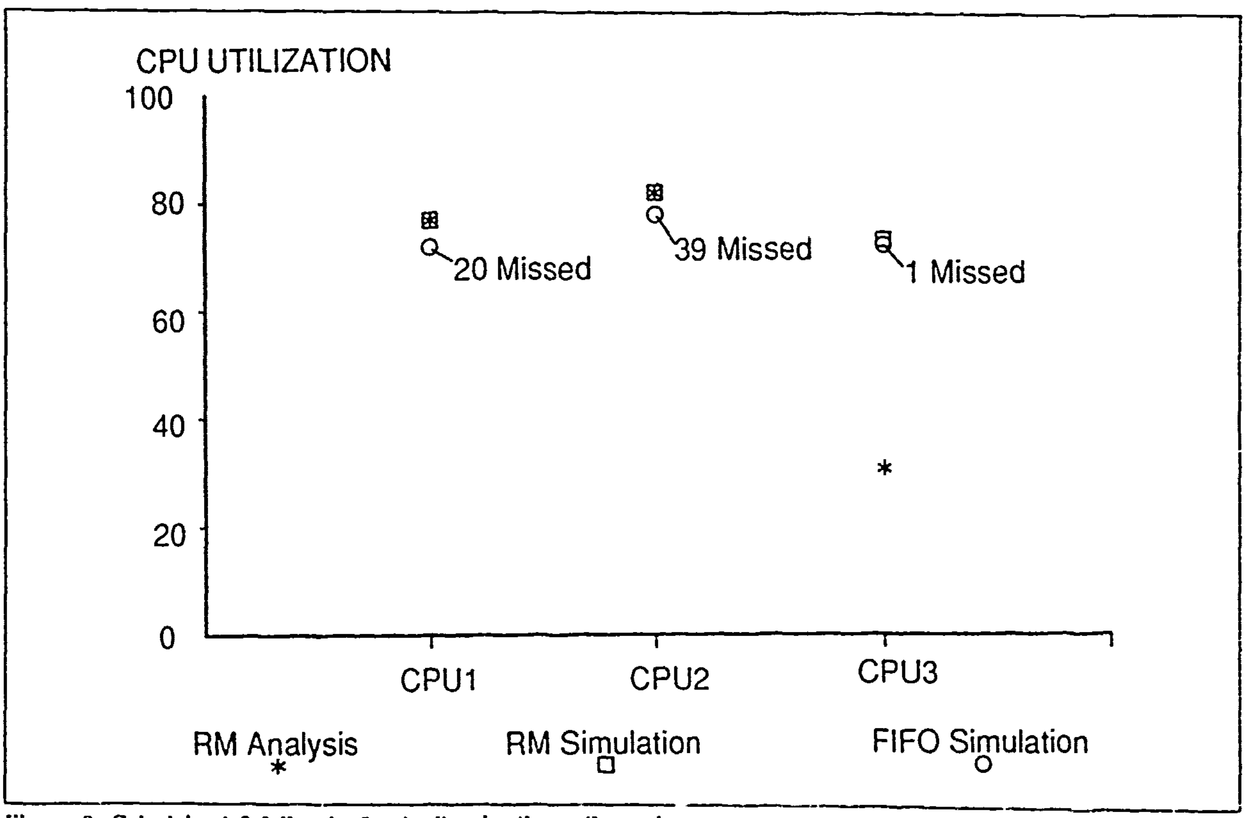

Figure 9. Scheduler 1-2-3 Results for the Passive Sonar Scenario

6.0 Summary

Three scenarios have been described and an architecture for one has been defined and analyzed. Each is functionally similar to what can be expected in future Navy systems. Each has been parameterized so that the required processing resources can be scaled up or down. The spreadsheet contains nominal values chosen to produce a moderate load on a testbed of three, 3 MIP machines. Our analysis for scenario 1 was based on having three 1 MIP machines. The parameters were adjusted accordingly so that the utilizations did not exceed 100% and the task set was schedulable.

The resources required to support the Submarine Passive Sonar scenario depend on the number of hydrophones and the number and type of formed beams. The more phones and beams, the more resources required. The resources required to carry out the functions of the Submarine CIC scenario depend on the number of operators. The Surface Ship Radar scenario resources depend on the number of tracks held. The spreadsheets referred to in this paper may be obtained from Dr. Jane Lui at the Univerity of Illinois, Urbana. As experience is gained in using these scenarios we anticipate that they will be refined and other interesting examples will be generated by the research community.

7.0 References

1 H.J. Currow and B.A. Wichmann, "A Synthetic Benchmark" Computer Journal 19(1):48-49, January 1976.

2 C.J. Date, "An Introduction to Database Systems Vol II", Addison-Wesley 1983.

3 W.C.Knight et al, "Digital Signal Processing for Sonar", IEEE Proceedings, vol. 69, Nov 1981.

4 C.L. Liu and J.W. Layland, "Scheduling Algorithms for Multiprogramming in a Hard Real Time Environment" JACM 20(1):46-61, 1973.

5 David T. Marlowe "Requirements for a High Performance Transport Protocol for Use on Naval Platforms", Report No. ANSI X3S3.3 HSP-8, July 23, 1989.

6 J.L. McClane, "The Ticonderoga Story", U. S. Naval Institute Proceeding, May 1985

7 D. L. Mills, "Network Time Protocol (Version 2) Specification and Implementation", DARPA Network Group Report RFC-1119, September 1989.

8 S. J. Rabinowitz, et al, "Applications of Digital Technology to Radar", Proceedings of the IEEE, vol. 73, Feb. 1985.

9 H.Tokuda, C.W.Mercer and Y.Ishikawa, "The ARTS Distributed Real-Time Kernel and its Toolset".

10 A.A. Winder, "II. Sonar System Technology", IEEE Trans. Sonics Ultrasonics, vol. SU-22, Sept 1975.